PHASE 4

AIR QUALITY MONITORING

OBJECTIVE:

The Objective of this IOT project is to monitor the Air Quality by measuring the AQI(Air Quality Index) value based only on particulate matter.

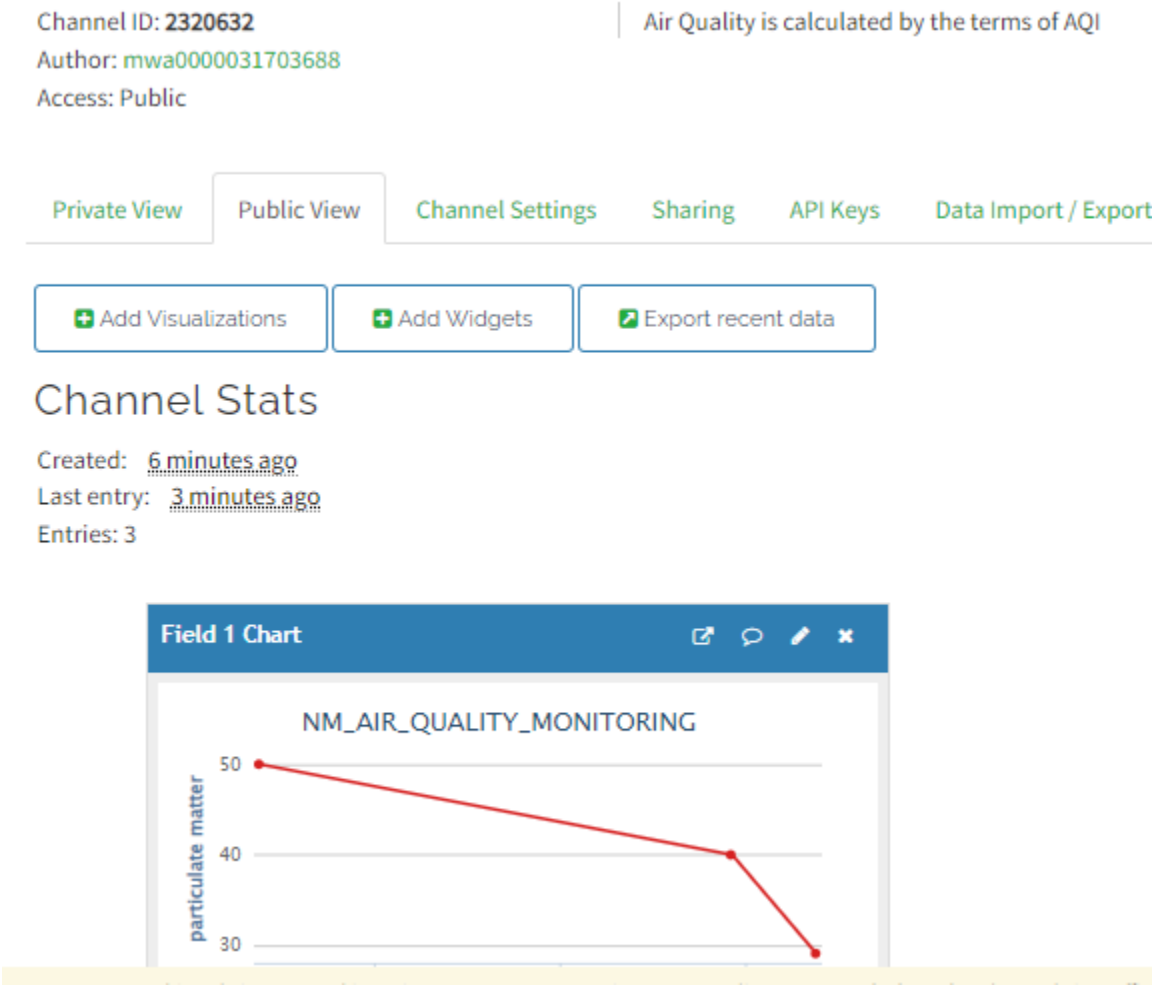
EXPLANATION:

- In the previous phase,we have done a half part of our project.
- So far we have done the transmission of particulate matter value from the PM2.5 sensor to the Thingspeak platform by using Arduino and ESP8266 Wifi Chip.
- In this phase,we are designing the user interface which displays the air quality by measuring the AQI(Air Quality Index) value.
- AQI value has been calculated by receiving the particulate matter value from the Thingspeak platform.
- By AQI value the Air quality has been classified and displayed as very good,good,fair,poor,very poor and hazardous.

AQI	Description
0–33	Very Good
34–66	Good
67–99	Fair
100–149	Poor
150–200	Very Poor
200+	Hazardous

- By providing an example value to test the web application to retrieve value from Thingspeak.

- For that we need to create channels and data should be provided.



CODE(HTML,CSS,JAVASCRIPT):

```
<!DOCTYPE html>
<html>

<head>
  <title>ThingSpeak Data Validation</title>
  <style>
    .center {
      margin: auto;
      width: 80%;
      padding: 20px;
      background-color: #f3f3f3;
      border-radius: 10px;
      text-align: center;
      box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    }

    h2 {
      color: #4e7cad;
    }

    #val1 {
      font-style: italic;
      font-size: 20px;
      margin: 20px;
    }

    #data-validation {
      font-style: italic;
```

```
      font-size: 18px;
      margin: 20px;
      color: #d62020;
      /* Red color for the status message */
    }

    iframe {
      border: none;
    }
  </style>
</head>

<body>
  <div class="center">
    <h2>AIR QUALITY DETECTION</h2>
    <div id="val1" style="font-style: italic;"></div>

    <p><strong>STATUS:</strong></p>
    <p id="data-validation" style="font-style: italic;"></p>
    <iframe width="450" height="260" style="border: 1px solid #cccccc;"

src="https://thingspeak.com/channels/2320632/charts/1?bgcolor=%23ffffff&color=%2
3d62020&dynamic=true&results=60&type=line&update=15"></iframe>
  </div>

  <script>
    const channelId = '2320632'; // Replace with your ThingSpeak channel ID
    const apiKey = 'Y44LFH2TF2O0W6UR'; // Replace with your ThingSpeak read API key
```

```
    const url =
`https://api.thingspeak.com/channels/${channelId}/feeds.json?api_key=${apiKey}&resu
lts=1`;

    fetch(url)
      .then(response => response.json())
      .then(data => {
        const pm25Value = parseFloat(data.feeds[0].field1);



        let Message, value;
        if (pm25Value >= 0 && pm25Value <= 30) {
          value = `Particulate matter value:${pm25Value}`
          Message = `very good`;
        } else if (pm25Value >= 31 && pm25Value <= 60) {
          value = `Particulate matter value:${pm25Value}`
          Message = `good`;
        } else if (pm25Value >= 61 && pm25Value <= 90) {
          value = `Particulate matter value:${pm25Value}`
          Message = `fair`;
        } else if (pm25Value >= 91 && pm25Value <= 120) {
          value = `Particulate matter value:${pm25Value}`
          Message = `poor`;
        } else if (pm25Value >= 121 && pm25Value <= 250) {
          value = `Particulate matter value:${pm25Value}`
          Message = `very poor`;
        } else {
          value = `Particulate matter value:${pm25Value}`
          Message = `hazardous`;
        }
```

```
      const val = document.getElementById('val1');
      val.textContent = value;
      const validationDiv = document.getElementById('data-validation');
      validationDiv.textContent = Message;
    })
    .catch(error => console.error('Error fetching data:', error));
 </script>
</body>

</html>
```

WEB APPLICATION: